ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015.

Task Scheduling in Federated Cloud by Multilevel Queue based on Genetic Algorithm

Dr.M.Srivenkatesh[1], K,Vanitha[2]

Associate Professor, GITAM Institute of Technology, GITAM University, Andhra Pradesh, India[1]

Assistant Professor, Department of Computer Science, GITAM Institute of Science, Andhra Pradesh, India[2]

Abstract: Federated Cloud computing is one of the upcoming latest new computing paradigm where applications and data services are provided by multiple cloud providers over the Internet. Today's some of the business organizations use federated cloud environment. The Task management is the important role in federated cloud computing systems. Task scheduling problems are which relate to the efficiency of the whole federated cloud computing facilities. The main focus is to study various problems, issues and types of scheduling in cloud computing. We propose Multi level Queue Task Scheduling Algorithm includes proposed Priority Based Round-Robin Task scheduling algorithm and task scheduling based on genetic algorithm for federated cloud computing environment.

Keywords: Cloud computing, dynamic scheduling, Genetic Algorithm, optimization.

I. INTRODUCTION

1. Federated Cloud Computing

They are some deifications of cloud computing namely Inherently Limited Scalability of Single-Provider Clouds, Lack of Interoperability among Cloud Providers. No Built-In Business Service Management Support .To address these issues Federated Cloud Computing was introduced. Cloud federation brings together different service providers and their offered services so that many Cloud variants can be tailored to match different sets of customer requirements. Cloud provider can provide resources to satisfy complex application request only if he holds infinite resources at his premises. Since this is not the case, so providers need to collaborate to be able to fulfill requests during peak demands and negotiate the use of idle resources with other peers. A federated cloud (also called cloud federation) is the deployment and management of multiple external and internal cloud computing services to match business needs. A federation is the union of several smaller parts that perform a common action. Scalability-- Cloud bursting to address peak demands .The major advantages of federated cloud computing is

- Scalability-- Cloud bursting to address peak demands
- Collaboration--Sharing of infrastructure between partners
- Multi-site Deployments-- Infrastructure aggregation across distributed data centers
- Reliability--Fault tolerance architectures across sites
- Performance--Deployment of services closer to end users
- Cost--Dynamic placement to reduce the overall infrastructure cost
- Energy Consumption--Minimize energy consumption.

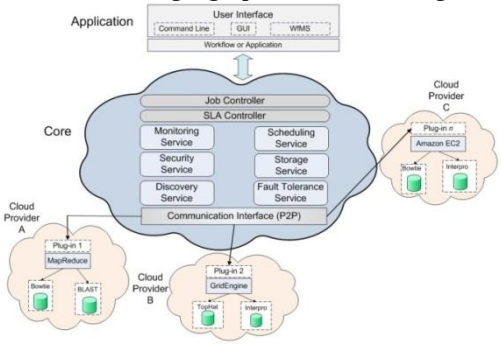
Federated clouds, by providing end to end quality of services, offer many advantages over traditional cloud services, which are:

Guaranteed performance: Due to limited resources, that are available with a single cloud service provider, sudden increase in workload may lead to deterioration of performance. Cloud federation overcomes this disadvantage by hiring resources from foreign cloud service providers, thereby guaranteeing the agreed Quality of Service. Also, high priority processing is guaranteed by delegating low priority processing tasks to foreign cloud service providers.

Guaranteed availability: During unexpected disasters, the cloud system will be able to recover the services by federating with other cloud service providers in unaffected areas. Availability may be guaranteed according to the priority of the service, as disaster recovery may not be an instant process.

Convenience of service cooperation: Cloud federations greatly increase the convenience by providing a one stop solution such that the consumer can see all the services together. For example, while applying for a passport, all the associated services may be integrated as one single service.

Dynamic load distribution: Geographical distribution of clients for every cloud service provider is highly uneven. In order to provide seamless services, dynamic load distribution is facilitated by cloud federations so that they could rise above their geographical shortcomings.



DOI 10.17148/IJARCCE.2015.43106

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015.

2. TASK SCHEDULING IN FEDERATED CLOUD COMPUTING

In the industry, Task scheduling is considered as one of the most famous combinatorial optimization problems. The main goal is to determine a proper sequence where tasks are executed while obeying to some (transaction logic) constraints. Implementations are labeled either as centralized or decentralized, static or dynamic, or a hybrid. All with their own strengths and limitations. The Task management is the key role in federated cloud computing systems. Task scheduling problems are main which relate to the efficiency of the whole federated cloud computing facilities. Genetic algorithm to be developed for the scheduler do the task scheduling, then execute the task.
The rest of the paper is organized as follows: Section 1 deals with a brief section presenting introduction to federated cloud computing and its importance of Task scheduling in section2, Section 3 presents the State of Art (Literature Review). Section 4 deals with Scheduling Techniques in Federated Cloud Computing and its benefits and proposed multilevel queue scheduling and proposed priority Based Round-Robin Task scheduling algorithm . Need for genetic Algorithms Section described 5. Section 6 presents proposed genetic algorithm for task scheduling conclusion and future lines of work in Section 7 and references are in section 8.

3. STATE OF THE ART

In the cloud computing environment, we have several task scheduling algorithms based on genetic algorithms.

The idea for generating initial population by using the Min-Min and Max-Min techniques for Genetic Algorithms as described in [1].

The task scheduler calls the Genetic Algorithm scheduling function for every task scheduling cycle ,this function creates a set of task schedules and evaluates the quality of each task schedule with user satisfaction and virtual machine availability and the function iterates genetic operations to make an optimal presented in [2].

In another investigation [3]. in dynamic scheduling task arrival is uncertain at run time and allocating resources are tedious as all tasks arrive at the same time, to avoid this genetic algorithm is used. Genetic algorithm is a heuristic method that deals with the natural selection of solution from all possible solutions and tasks are scheduled according to the computation and memory usage was presented.

To find a fast mapping using genetic algorithms with exist if satisfy condition to speed up the mapping process and ensures the respecting of all task deadlines as described in [4].

Another investigation[5], characteristics of a private cloud used for e-Learning purposes along with a genetic algorithm that optimizes the scheduling of the e-Learning workloads according to a set of conditions that are imposed by the underlying virtualization technology such as memory over commitment and IOPS rate was discussed .

The VM resources scheduling in cloud computing environment and with the advantage of genetic algorithm, balanced scheduling strategy of VM resources based on genetic algorithm as presented in [6].

Real-time load parameters from the server cluster nodes as decision variables of resources scheduling model, and uses the improved adaptive genetic algorithm, in order to realize the load balancing scheduling of cloud resource, and make the each index change smoothly as presented in [7].

We have interesting investigation on GA Module Queue Sequencer, Job Scheduler (JS) and Resource Pool (RP) and all service requests which are coming from Cloud Users domain are stored in RQ which is in GAQS. The requested processes must communicate with GAQS processor (GAP) and the processor finds out the appropriate sequence of tasks which reduce the waiting time of the tasks. GAQS processor then communicate directly with JS which schedules the tasks using Round Robin scheduling algorithm and communicate with RP and tries to assign each of these jobs as per their requirement to the resources [8].

In order to improve private cloud environment ,genetic algorithms are proposed with combination of SCFP (Shortest Cloudlet to Fastest Processor), LCFP (Longest Cloudlet to Fastest Processor) and a meta-heuristic GA as an optimization method, developed a new approach Modified Genetic Algorithm (MGA) for task scheduling. MGA is developed by modifying the initial population with LCFP, SCFP and by controlling the stochastic operators of standard genetic algorithm which lead to achieve a very good results and better efficiency of the algorithm than the standard genetic algorithm [9]
.
A macroscopic scheduling model with cognition and decision components for the cloud computing, which considers both the requirements of different jobs and the circumstances of computing infrastructure, proposed a job scheduling algorithm based on Multi-Objective Genetic Algorithm (MO-GA), taking into account of the energy consumption and the profits of the service providers, and providing a dynamic selection mechanism of the most suitable scheduling scheme for users according to the real time requirements as presented in [10].

We have resource optimization within a cloud, a Hybrid Genetic Algorithm (HGA for short in the following), which acts as an independent module in the cloud manager. It considers CPU load, network throughput and disk I/O load of all the virtual machines carried on one specific physical machine, can be matched and calculated to get the optimal migration advice and proposed genetic

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015.

algorithm with multiple fitness, and three sub fitness functions namely the virtual machines'load complementation, the amount of physical machines is minimum after migration, that the amount of virtual machines that need to be migrated described in [11].

4. SCHEDULING TECHNIQUES

4. **Scheduling in federated cloud computing**

All the clients are to be received by cloud resource broker. After receiving the entire service request, it sends the received request to scheduler. Scheduler has to accurately calculate the task priority of each client task. In genetic algorithms we consider both request size and request time also. User sends a request to the cloud broker. After receiving the request from the user, the cloud broker forwards the request to the cloud. Cloud has the CSP (Cloud Service Provider) after receiving the request the CSP forwards to Scheduler. The Scheduler will receive the task and based on the priority based round robin algorithm the client task has to be executed first. The requested data has to be sent to the appropriate user.

4.1. Proposed Multi level Queue Task Scheduling Algorithm

Ready queue which contains tasks are partitioned into two separate queues namely foreground(Interactive) and background (batch).Each queue has its own scheduling algorithms, foreground Priority based round robin task scheduling and background first come first served algorithm are followed. Scheduling is done between the queues.

1. Fixed priority scheduling it serves all from foreground then from background. it may have possibility of starvation.
2. Time slice –Each queue gets a certain amount of resource time which can be schedule among its tasks. That is 80% to foreground in Priority based Round Robin task scheduling and 20% to background in first come first served algorithm.

Consider five tasks viz. A, B, C, D ,E,F and ,G with g i v e n resource execution time .

Serial Number	Task	Execution Time
1	A	22
2	B	18
3	C	9
4	D	10
5	E	4
6	F	5
7	G	3

Table I. Input component for the Tasks

4.2. Proposed Priority based Round Robin Task Scheduling

The algorithm performs following steps:

Step 1: Allocate Resource to every task in round robin fashion, according to the given priority, for given time quantum (say k units) only for one time.

Step 2: After completion of first step following steps are performed:

a) Tasks are arranged in increasing order or their remaining resource execution time in the ready queue. New priorities are assigned according to the remaining resource execution of tasks; the tasks with shortest remaining resource execution time are assigned with highest priority.

b) The Tasks are executed according to the new priorities based on the remaining resource execution time and each task gets the control of the required resource until they finished their execution.

Case I:

Consider only five tasks viz. A, B, C, D and E with g i v e n resource execution time and associated priorities. Let the time quantum is 5 ms. The 80% of Tasks are considered from table one.

Serial Number	Task	Execution Time	Priority
1	A	22	4
2	B	18	2
3	C	9	1
4	D	10	3
5	E	4	5

Table II. Input component for the Tasks

According to proposed algorithm:-

Priority based Round Robin Task Scheduling consists of two rounds:

Round 1: Tasks with the highest priority is executed first for the time equal to given time quantum i.e. 5 ms. In the same manner other tasks are executed according to their priorities for single time quantum. Eg: The sequence of execution for above case is:

Serial Number	Task	Execution Time	Priority
1	C	5	1
2	B	5	2
3	D	5	3
4	A	5	4
5	E	4	5

Table III. Executed Task Execution for first round

Round 2: This round includes the changing of Tasks's priorities according to the remaining resource execution time. The task with least remaining resource execution time is assigned highest priority. The new assigned priorities are as follows:

Serial Number	Task	Remaining Execution Time	Priority
1	C	4	1
2	D	5	2
3	B	13	3
4	A	17	4

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015.

Table IV. Remaining resource Execution for the second round & New assigned priorities

Now the Tasks are executed according to the new priority assigned without taking consideration of time quantum.
Average waiting Time: 26.200001 ms
Average Turnaround Time: 38.800000 ms

Case II:
Consider the same problem with varying time quantum.
Let the time quantum is 9 ms.
According to our algorithm:-
Execution takes place in two rounds:
Round 1: Tasks with highest priority is executed first for the time equal to given time quantum i.e. 9 ms. In the same manner other Tasks are executed according to their priorities for single time quantum of execution for above case is:

Serial Number	Task	Execution Time	Priority
1	C	9	1
2	B	9	2
3	D	9	3
4	A	9	4
5	E	4	5

Table V. Executed Task Execution for first round

Round 2: This round includes the changing of Tasks's priorities according to the remaining resource execution time. The task with least remaining resource execution time i.e D(1 ms) is assigned highest priority. The new assigned priorities are as follows:

Serial Number	Task	Remaining Execution Time	Priority
1	D	1	1
2	B	9	2
3	A	13	3

Table VI. Remaining resource Execution for the second round & New assigned priorities

Now the Tasks are executed according to the new priority assigned without taking consideration of time quantum.
Average waiting Time: 28.000000 ms
Average Turnaround Time: 40.600000 ms
First Come First Served Algorithm for remaining tasks
For the remaining task first come first served algorithm is followed. Task F is allocated with required resource first and task G is allocated the required resource..

Serial Number	Task	Execution Time
6	F	5
7	G	3

Table VII. Input component for the Tasks

Average waiting Time: 6.00 ms

5. GENETIC ALGORITHM

Genetic algorithm is a method of scheduling in which the tasks are assigned resources according to individual solutions (which are called schedules in context of scheduling), which tells about which resource is to be assigned to which task.
Genetic Algorithm is based on the biological concept of population generation. The main terminology used in genetic algorithms is:

- Individual: Any Possible Solution.
- Population: Group of all individuals.
- Search Space: All possible solutions to the search problem.
- Chromosome :Blue print for an Individual or population of abstract representations or Genotype or Genome
- Trait: Possible aspect (feature) of an Individual.
- Allele: Possible settings of Trait.
- Locus: The Position of a gene on the chromosome
- Genome: Collection of all Chromosome for an individual
- Genotype: Particular set of genes in genome
- Phenotype: Physical Characteristics of genotype
- Chromosomes could be Bit Strings, Real Numbers, Permutation of Element, List of Rules, Program Elements (Generic Programming), Any Data Structure etc

The main terms used in genetic algorithms are:

i). Initial Population

Initial population is the set of all the individuals that are used in the genetic algorithm to find out the optimal solution. Every solution in the population is called as an individual. And every individual is represented as a chromosome for making it suitable for the genetic operations. From the initial population the individuals are selected and some operations are applied on those to form the next generation. The mating chromosomes are selected based on some specific criteria. Chromosomes could be Bit Strings, Real Numbers, Permutation of Element, List of Rules, Program Elements (Generic Programming), Any Data Structure etc

ii). Fitness Function

A fitness function is used to measure the quality of the individuals in the population according to the given optimization objective. The fitness function is defined over the genetic representation and measures the *quality* of the represented solution. The fitness function is always problem dependent.

iii). Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a *fitness-based* process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015.

only a random sample of the population, as this process may be very time-consuming. Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions

iv. Crossover

We have different cross over like single point cross over, Two Point cross over, uniform cross over, and Arithmetic cross over We use single point cross over operator. In single point cross over, binary string from beginning of chromosome to the cross over point is copied from one parent, the rest is copied from second parent

v. Mutation

After selection and crossover, you now have a new population full of individuals. Some are directly copied, and others are produced by crossover. In order to ensure that the individuals are not all exactly the same, you allow for a small chance of mutation. You loop through all the alleles of all the individuals, and if that allele is selected for mutation, you can either change it by a small amount or replace it with a new value. The probability of mutation is usually between 1 and 2 tenths of a percent. Mutation is fairly simple. You just change the selected alleles based on what you feel is necessary and move on. Mutation is, however, vital to ensuring genetic diversity within the population.

Genetic Algorithm works in the following manner:

1. Begin
2. Initialize population with random candidate solutions
3. Evaluate each candidate
4. Repeat Until (termination condition is satisfied)
    a. Select parents
    b. Recombine pairs of parents
    c. Mutate the resulting offsprings
    d. Evaluate new candidate
    e. Select individuals for the next generation;
5. End

6. PROPOSED GENETIC ALGORITHM FOR TASK SCHEDULING

In Genetic Algorithm the initial population is generated randomly, so the different schedules are not so much fit, so when these schedules are further mutated with each other, there are very much less chances that they will produce better child than themselves. We have provided an idea for generating initial population by using the Multi level Queue Task scheduling using Genetic Algorithms.

As discussed in Genetic Algorithm; the solutions that are fit, give the better generations further when we apply genetic operators on them and hence if multi level queue task scheduling algorithm will be used for the individual generation, we will get the better initial population and further the better solutions than in the case of standard Genetic Algorithm in which initial population is chosen randomly. The new Improved Genetic Algorithm is like given below:

1. Begin
2. Find out the solution by Multi level Queue Task scheduling algorithm
3. Initialize population by the result of Step 2
4. Evaluate each candidate
5. Repeat Until (termination condition occur)
    a. Select parents
    b. Recombine pairs of parents
    c. Mutate the resulting offsprings
    d. Evaluate new candidate
    e. Select individuals for next generation
6. End

REFERENCES

[1]. Sung Ho Jang, Tae Young Kim, Jae Kwon Kim and Jong Sik Lee, *The Study of Genetic Algorithm-based Task Scheduling for Cloud Computing School of Information Engineering, Inha University* #253, YongHyun-Dong, Nam-Ku

[2]. A.Kaleeswaran1,V.Ramasamy2,P.Vivekanandan3, *Dynamic scheduling of data using genetic algorithm in cloud computing*, Park College of Engineering and Technology, Coimbatore, India

[3]. Sandeep Tayal1,*Tasks Scheduling optimization for the Cloud Computing Systems, University Schoolof Information Technology* Guru Gobind SinghIndraprastha University, Delhi-110006,India]

[4]. Ali Mamat, Hamidah Ibrahim and Shamala K. Subramaniam ,*Impatient Task Mapping in Elastic Cloud using Genetic Algorithm Nawfal A. Mehdi*, Department of Computer Science,Faculty of Computer Science and Information Technology, University Putra Malaysia

[5]. Romania octavian.morariuPolitehnica Bucharest, *A Genetic Algorithm for Workload Scheduling In Cloud Based e-Learning*, Octavian Morariu University

[6]. JianhuaGuJinhua Hu Tianhai Zhao Guofei, *A New Resource Scheduling Strategy Based on Genetic Algorithm in Cloud Computing Environment*, Sun School of Computer

[7]. XinLUa;_, Jing ZHOUa, Dong LIU ba, *A Method of Cloud Resource Load Balancing Scheduling Based on Improved Adaptive Genetic Algorithm, School of Computer Science and Engineering*, University of Electronic Science and Technology of China

[8]. Sourav Banerjee, MainakAdhikary, UtpalBiswas, *Advanced Task Scheduling for Cloud Service Provider Using GenetaicAlgorithm*,Department of Computer Science and Engineering Kalyani Government Engineering College

[9]. ShaminderKaur, AmandeepVerma, *An Efficient Approach to Genetic Algorithm for Task Scheduling in Cloud Computing Environment*, Department of IT, UIET, Panjab University, Chandigarh, India

[10]. Jing Liu*1, Xing-Guo Luo2, Xing-Ming Zhang3, Fan Zhang4 and Bai-Nan Li5, *Job Scheduling Model for Cloud Computing Based on Multi-Objective Genetic Algorithm*,1, 2, 3, 4, 5 National Digital Switching System Engineering & Technology Research Center, Zhengzhou 450002, China.

[11]. Shi Chen, Jie Wu, Zhihui Lu , *A Cloud Computing Resource Scheduling Policy Based on Genetic Algorithm with Multiple Fitness*, School of Computer Science and TechnologyFudan University Shanghai, China, 200433+